International Journal of Neural Systems, Vol. 10, No. 2 (April, 2000) 79–93

A SEMI-PARAMETRIC HYBRID NEURAL MODEL FOR NONLINEAR BLIND SIGNAL SEPARATION

HANCHUAN PENG[*‡], ZHERU CHI[†] and WANCHI SIU
Center for Multimedia Signal Processing,
Department of Electronic & Information Engineering,
The Hong Kong Polytechnic University,
Hung Hom, Kowloon, Hong Kong
[*] *E-mail: phc@eie.polyu.edu.hk*
[†] *E-mail: enzheru@polyu.edu.hk*
[‡] *Department of Biomedical Engineering,*
Southeast University, Nanjing, 210096, China,
E-mail: phc@seu.edu.cn

Received 17 January 2000
Revised 20 March 2000
Accepted 27 March 2000

Nonlinear blind signal separation is an important but rather difficult problem. Any general nonlinear independent component analysis algorithm for such a problem should specify which solution it tries to find. Several recent neural networks for separating the post nonlinear blind mixtures are limited to the diagonal nonlinearity, where there is no cross-channel nonlinearity. In this paper, a new semi-parametric hybrid neural network is proposed to separate the post nonlinearly mixed blind signals where cross-channel disturbance is included. This hybrid network consists of two cascading modules, which are a neural nonlinear module for approximating the post nonlinearity and a linear module for separating the predicted linear blind mixtures. The nonlinear module is a semi-parametric expansion made up of two sub-networks, one of which is a linear model and the other of which is a three-layer perceptron. These two sub-networks together produce a "weak" nonlinear operator and can approach relatively strong nonlinearity by tuning parameters. A batch learning algorithm based on the entropy maximization and the gradient descent method is deduced. This model is successfully applied to a blind signal separation problem with two sources. Our simulation results indicate that this hybrid model can effectively approach the cross-channel post nonlinearity and achieve a good visual quality as well as a high signal-to-noise ratio in some cases.

1. Introduction

Recently, Blind Signal Separation (BSS) has drawn great attention in separating the statistically independent non-Gaussian sources.[1] Independent Component Analysis (ICA) is a popular method for BSS when these sources are mixed linearly.[1] For the following equation:

$$s = Ax \tag{1}$$

where $s = [s_1, \ldots, s_n]^T$ is the n-dimension observed signal, $x = [x_1, \ldots, x_n]^T$ stands for n independent sources $\{x_1, \ldots, x_n\}$ which are mixed with an unknown n by n linear matrix A, the linear BSS problem is to estimate x and a linear de-mixing matrix $W = A^{-1}$ from s, as shown in the following equation:

$$x = Ws \tag{2}$$

The solution is unique up to some trivial indeterminacies, including permutation and multiplication of s_i by constants.

Because the signals nonlinearly transformed from independent sources are still independent with each other, in principle, it is impossible to accurately restore the original independent sources or their whitened signals merely from the observed nonlinearly mixed signals. In other words, the nonlinear ICA does not have a unique solution.[2] Despite this limitation, recently there are some discussions on how to use neural networks to separate post nonlinear blind mixtures.[3–9] In the general form of post nonlinear ICA, x and W are estimated from a middle signal z, which is nonlinearly transformed from s with nonlinear operator F, as shown in the following equation:

$$x = Wz = WF(s) \tag{3}$$

In Taleb and Jutten's model, F is constrained to a nonlinear diagonal operator $F = \mathrm{diag}(f_1, \ldots, f_n)$, which does not include the cross-channel disturbance.[8] Its paradigm can be depicted as in Fig. 1. Yang, Amari, and Cichocki[9] proposed two information back-propagation training algorithms for this model. Two major problems of such a model are (1) the diagonal nonlinearity is not widely accepted because usually there is cross-channel nonlinear disturbance in signal transmission, and (2) the back-propagation-like algorithm often converges slowly and unsteadily due to the gradient descent strategy.

In this paper, we discuss how to "learn" an acceptable solution for post nonlinear ICA problems with cross-channel disturbance. Because there are an infinite number of possible solutions for such a problem, we propose a weak nonlinearity assumption to specify what the model should find. Our idea is implemented as a hybrid neural network that combines a nonlinearity approximation module and a linear ICA module. The nonlinearity approximation module is designed as a semi-parametric model, which is a weak nonlinear expansion from a linear neural network. The batch-learning algorithm based on the entropy maximization is derived. Our model is applied to separating the nonlinear blind mixtures, where special types of nonlinearity are designed. The paper is organized as follows. In Sec. 2, we present our hybrid model for nonlinear blind signal separation. The learning algorithm based on the entropy maximization for such a model is derived in Sec. 3. In Sec. 4, we report experimental results and discuss the performance of our model. Finally, concluding remarks are drawn in Sec. 5.

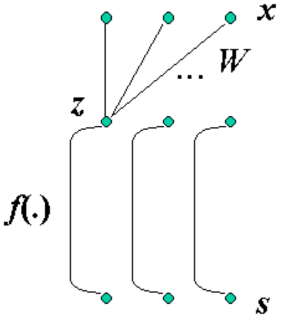


Fig. 1. Blind de-mixer for post diagonal nonlinearity.

2. Hybrid Neural Network Model

Different from the blind de-mixer shown in Fig. 1, which only approximates the diagonal nonlinear operator, a neural blind de-mixer consisting of two modules as shown in Fig. 2 is proposed. The first module is a nonlinear module to approximate the post nonlinearity. The second module is a linear ICA module to estimate the final restored signals.

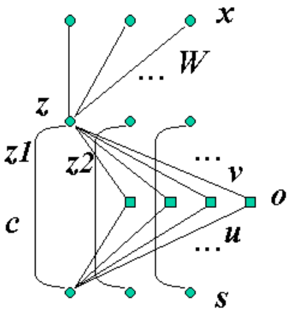


Fig. 2. Blind de-mixer with two sub-networks for post cross-channel nonlinearity.

2.1. *Nonlinear module*

For the nonlinear module, the post nonlinearity operator F is assumed as a "weak" operator that approaches relatively strong nonlinearity by parameter tuning. This assumption can be written in the following semi-parametric form:

$$F(s) = K(s) + U(s) + \varepsilon \tag{4}$$

where $K(s)$ is the function of a known parametric model, $U(s)$ is an unknown smooth function (nonparametric model), and ε is an independent random error $\varepsilon \sim N(0, \tau^2)$, where τ is a positive scale parameter. Our assumption requires that the influence of U is trivial at the beginning (thus K plays the main role) and becomes stronger gradually until a specific ending condition is satisfied. With this assumption, F can represent a large variety of nonlinearities, including the diagonal nonlinearity, n by n nonlinear matrix operator, and other more complex nonlinearities.

In Fig. 2, we propose a simplified version of the semi-parametric model mentioned above, where the parametric model K is chosen as a linear channel model and the non-parametric model U is approximated by a three-layer feedforward neural network. This paradigm leads to two parallel sub-networks, the first of which has the following output:

$$z1 = K(s) = c \circ s \tag{5}$$

where c is a coefficient vector and $\circ$ stands for the Hadamard product. This network actually can be regarded as a separated part from the common linear ICA model because the latter has the indeterminations of permutation and amplitude.

The second sub-network is a nonlinear cross-channel model, which has the following output:

$$\begin{aligned} z2 = U(s) &= \Omega[v \times o - v^{(b)}] \\ &= \Omega[v \times \Phi(u \times s - u^{(b)}) - v^{(b)}] \end{aligned} \tag{6}$$

where u, v, $u^{(b)}$, $v^{(b)}$ are the weight matrices and bias vectors, $\times$ is the matrix multiplication, Ω and Φ are the nonlinear activation functions of the output layer and hidden layer, respectively. The second sub-network is a three-layer perceptron, which is a feedforward neural network widely used in approximating any nonlinearity when there are enough hidden neurons in the layer o. This sub-network is a typical non-parametric model because it does not require any prior knowledge about the cross-channel nonlinearity.

The total output of the nonlinear neural module is given by

$$z = z1 + z2 = F(s) \tag{7}$$

For the model expressed in Eq. (7), when the nonlinear operator F is continuous, it is possible to approximate the strong nonlinearity from the "weak" operator. Therefore, a strategy to "learn" (construct) a proper solution is adopted. The method employed in our algorithm is to initialize the weights of the second sub-network to be trivial values and enlarge these weights in the learning of the whole network. Notice that the output of the nonlinear module, z, is the input to the following linear module, as expressed in Eq. (3).

2.2. *Linear module*

In Fig. 2, a linear ICA module, which is described by Eq. (3), is attached to the nonlinear module. In principle, any linear ICA paradigms can be applied to this module, however, the fast ICA algorithm[10] is adopted to produce a fast estimation. These two cascaded modules adopt different algorithms and therefore, form a hybrid neural network.

3. Learning Algorithm

Various algorithms based on information back-propagation can be deduced for the hybrid model. Because the function of the model is to construct mutual independent sources (components) of x, the information constraints for linear ICA can be used. These constraints include information maximization,[11] entropy maximization,[12,13] maximum likelihood estimation,[14,15] higher-order moment and cumulants,[16] nonlinear Principle Component Analysis (PCA),[17] etc. For simplicity, we illustrate here only one set of equations deduced from the entropy maximization, which is among the most important approaches of ICA.

Entropy can be "maximized" merely when the variables are bounded. Notice that the signal x is not bounded, thus we maximize the entropy of y, whose ith component is defined as

$$y_i = \sigma(x_i) = \frac{1}{[1 + \exp(-x_i)]} \tag{8}$$

The entropy of y can be given by

$$H(y) = H(x) + E[\ln|\det(J_\sigma)|] \tag{9}$$

where $\det(J_\sigma)$ is the determinant of the Jacobian of the mapping function from x to y, and $E[\cdot]$ is the expectation.

Similarly we have

$$H(x) = H(z) + E[\ln|\det W|] \tag{10}$$

$$H(z) = H(s) + E[\ln|\det(J_G)|] \tag{11}$$

where W and J_G are Jacobians of the mapping functions from z to x and from s to z, respectively.

When Ω is chosen as pure linearity and Φ is chosen as the hyperbolic tangent sigmoid transfer function, the above relationships sum up to be:

$$H(y) = H(s) + E\left[\ln\left|\prod_{i=1}^{n}(1-x_i)x_i\right|\right] + E[\ln|\det W|] + E[\ln|\det(J_G)|] \tag{12}$$

where

$$J_G = \operatorname{diag}(c) + \frac{1}{2}\{v \times [(\Gamma_{m:1} - o \circ o) \times \Gamma_{1:m} \circ u\} \tag{13}$$

$\Gamma_{P:Q}$ is a P-row-Q-column matrix whose components are all 1, and m is the number of hidden neurons in the neural network. The first term [a diagonal matrix whose n elements are the corresponding elements of vector c in Eq. (5)] of Eq. (13) is the Jacobian from s to $z1$ (for the first sub-network of the nonlinear module). The second term (a n by n matrix) of Eq. (13) is the Jacobian from s to $z2$ (for the second sub-network of the nonlinear module). Due to Eq. (7), the Jacobian from s to z is the sum of these two terms.

Since that the task of the nonlinear module is to generate a coarse estimation of the linear mixture from the original nonlinear blind mixture, it is reasonable to force the coarse estimation z to be stable when the linear ICA model works. This requirement is combined with Eq. (12) to produce the following cost function for z:

$$C = \|z(k) - z(k-1)\| - \lambda H(y) \tag{14}$$

where k is the index of a loop, λ is a positive factor, and $\|\cdot\|$ is the 2-norm. Minimization of the first term of Eq. (14) will force the nonlinear module to produce slowly varying output, while maximization (minimization the negative) of the second term is a requirement to produce independent components. The maximization of the second term had been used similarly in the information back-propagation approach.[9]

The cost function in Eq. (14) is minimized to train the nonlinear module. Because the repeating adjustment for every time step is complicated and a significant part of such adjustment may not result in an immediate improvement of the estimation of z (this is a characteristic of the gradient descent algorithm!), we propose a batch learning algorithm to find the parameters of the nonlinear module. In this batch learning algorithm, all "micro" adjustments over a full loop sum up to act as a "macro" variation, which is used to update the parameters of the nonlinear module. Under the constraint of Eq. (14), the total gradient of C to z is approximated as (see Appendix A for the derivation):

$$\nabla_z \approx 2\{E[z(k)] - E[z(k-1)]\} - \lambda\left\{W^T \times \frac{\Gamma_{n:1} - 2E[x]}{E[x] - E[x \circ x]}\right\} \tag{15}$$

Back-propagation-like learning algorithm of the nonlinear module can be obtained straightforwardly when ∇_z is available. In the batch learning, the following approximation equations are used to adjust parameters of the nonlinear module (see Appendix B for derivation):

$$\Delta c = -\mu \nabla_z \circ E[s] \tag{16}$$

$$\Delta v = -\mu \nabla_z \times E[o]^T \tag{17}$$

$$\Delta v^{(b)} = \mu \nabla_z \tag{18}$$

$$\Delta u = -\frac{\mu}{2}\{v^T \times \nabla_z \circ (\Gamma_{m:1} - E[o \circ o])\} \times E[s]^T \tag{19}$$

$$\Delta u^{(b)} = \frac{\mu}{2}\{v^T \times \nabla_z \circ (\Gamma_{m:1} - E[o \circ o])\} \tag{20}$$

where μ is a small learning rate and a vanishing function of learning loop k (time).

Although there are two terms in Eq. (14), in our algorithm the ending condition is only dependent on the smoothness of the middle signal z. Two indexes are defined for this purpose:

$$e_1 = \|z(k) - z(k-1)\| \tag{21}$$

$$e_2 = \frac{\|z(k) - z(k-1)\|}{\|z(k)\|} \tag{22}$$

The first index, e_1, is actually the first term in Eq. (14). When this index is less than a given threshold, the learning of the nonlinear module should stop. Otherwise a new estimation of z is required. In the case that z is very close to zero, the second index, e_2, is used.

The complete batch learning algorithm is presented in Table 1. Notice that the linear ICA module and the nonlinear module are updated asynchronously.

Table 1. The batch learning algorithm of the hybrid model.

Step 1: $k = 0$. Initialize z to be the sum of s and small perturbation, that is, $c(k) = \Gamma_{n:1}$ and u, v, $u^{(b)}$, $v^{(b)}$ are randomly initialized to produce small $z2$. Calculate the expectation of s.

Step 2: $k = k + 1$. Call a linear ICA algorithm to de-mix z and produce independent sources x.

Step 3: Calculate the expectation of x.

Step 4: Calculate ∇_z according to Eq. (15).

Step 5: Calculate the expectations of z, o.

Step 6: Adjust the parameters c, u, v, $u^{(b)}$, $v^{(b)}$ according to Eqs. (16)–(20).

Step 7: Calculate z with the new parameters.

Step 8: Calculate one of the indexes in Eqs. (21) and (22), If the index is less than a preset small positive threshold, then stop the processing and output results. Otherwise go to Step 2.

4. Experimental Results and Discussion

For simplicity and convenience in comparing with the existing ICA and nonlinear ICA algorithms, we choose four types of nonlinearity for evaluation. The pixel intensity range of the original images (ground truth signals) is transformed to $[-1, 1]$. They are linearly mixed with a random mixing matrix A to generate the middle signal $h = [h_1, h_2]^T$, which are then nonlinearly transformed with the following relationships to produce the "observed" signal s:

$$\begin{cases} s_1 = \tanh(h_1) + \ln(1 + |h_1 + h_2|) \\ s_2 = \sinh(h_2) - \sqrt{|h_1 \circ h_2|} \end{cases} \tag{23}$$

The four nonlinear functions in Eq. (23) lead to obvious cross-channel nonlinear disturbances. These nonlinear disturbances can be visualized in Fig. 3, where s_1 and s_2 are complex curves. [The curves in

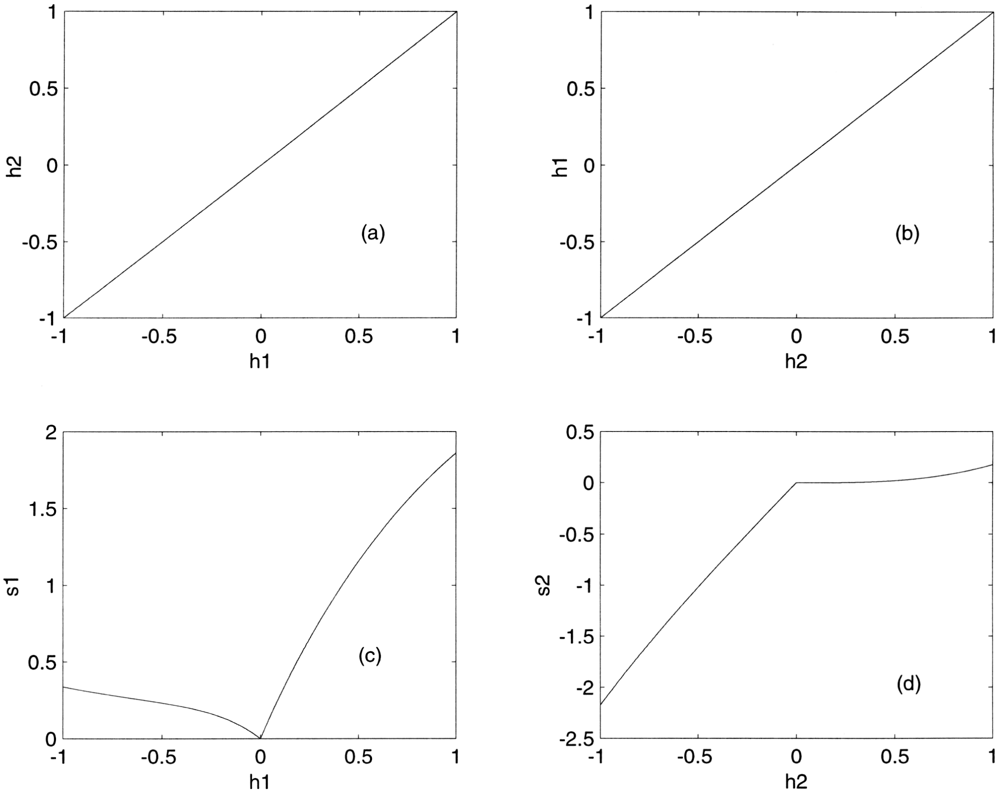


Fig. 3. Cross-channel nonlinear relationships of Eq. (23).

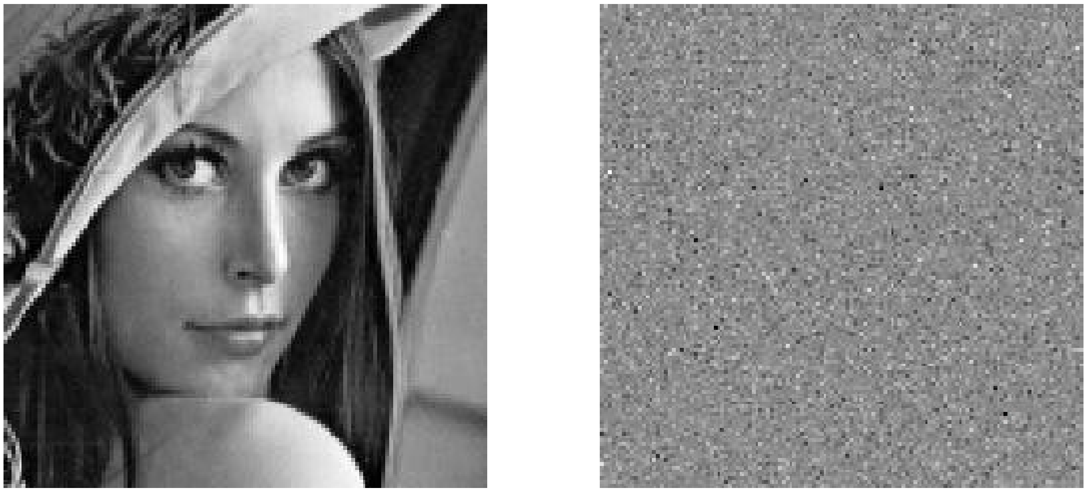
Figs. 3(c) and (d) are obtained when h_1 and h_2 have relationship in Figs. 3(a) and (b).]

The task of nonlinear ICA is to restore the ground truth signals from s. Two main experiments are designed to investigate the performance of our semi-parametric hybrid model.

4.1. *Experiment 1: Nonlinear ICA of image-noise mixtures*

For the convenience of visualization, we used 128 × 128 "Lenna" image as one source and the impulsive noise as another source (see Fig. 4). They are used as row-by-row 1-dimensonal signals in our experiments.

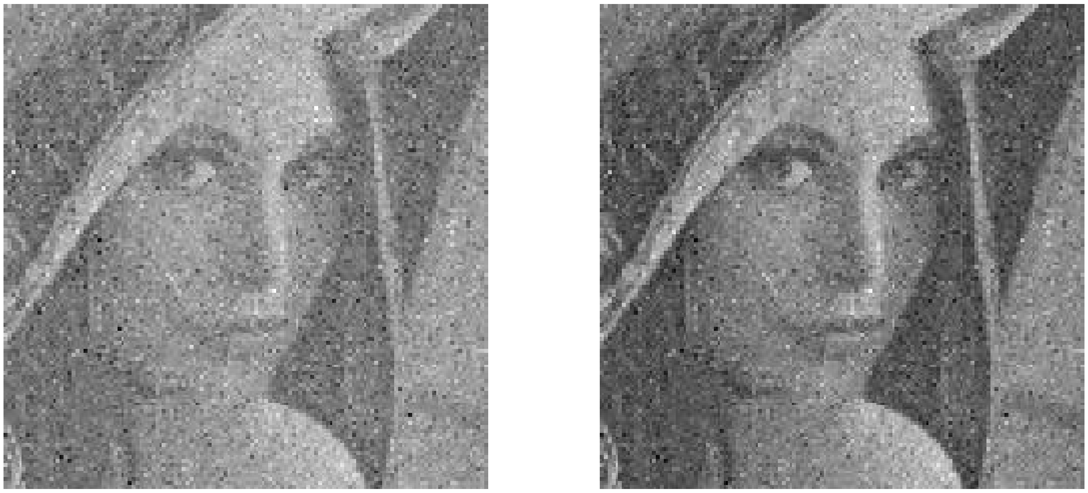
A randomly generated mixing matrix, A, is used to produce the linear mixed middle signal h. Because this experiment is repeated for a number of times, the real value of the mixing matrix is unimportant. For the following reported results, A is $\begin{bmatrix} 0.3077 & 0.7911 \\ 0.4893 & 0.8805 \end{bmatrix}$. Figures 5 and 6 show h and s, respectively. Then s is scaled to the range of $[-1, 1]$.



(a) Lenna

(b) Impulsive noise

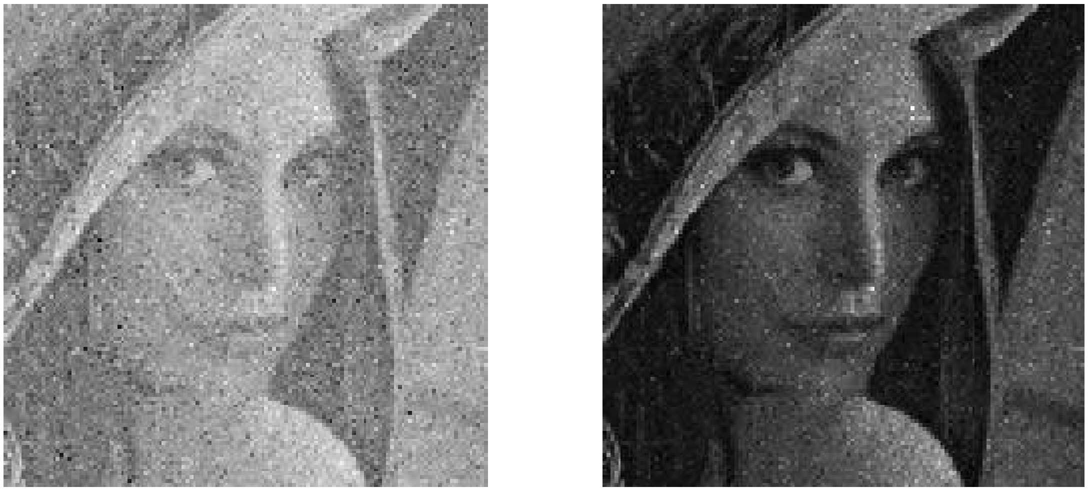
Fig. 4. The ground truth signals used in experiment 1.



(a) Channel 1

(b) Channel 2

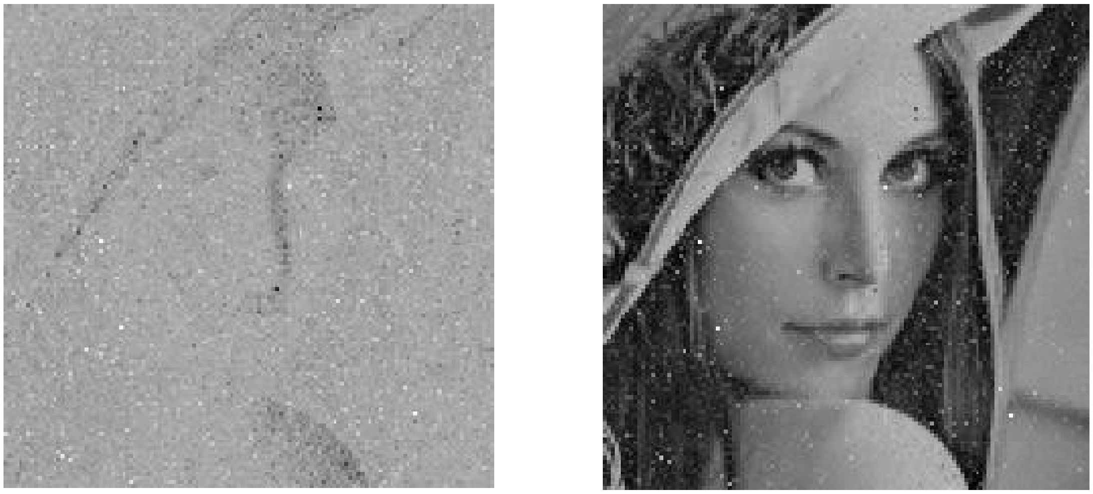
Fig. 5. The linear mixed signal h in experiment 1.



(a) Channel 1

(b) Channel 2

Fig. 6. The nonlinear transformed signal s in experiment 1.

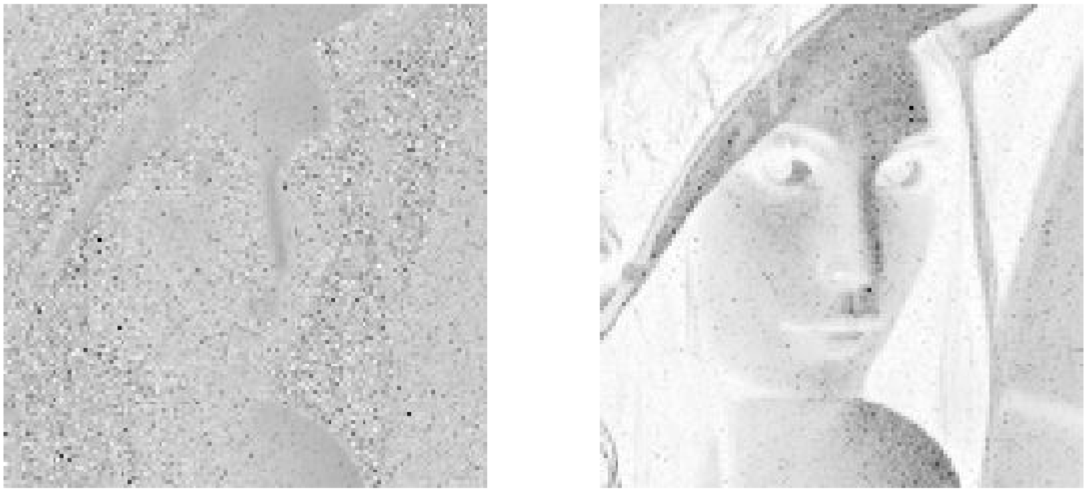


(a) Channel 1

(b) Channel 2

Fig. 7. The nonlinearly restored signal x from the hybrid model using both the first sub-network and the second sub-network.

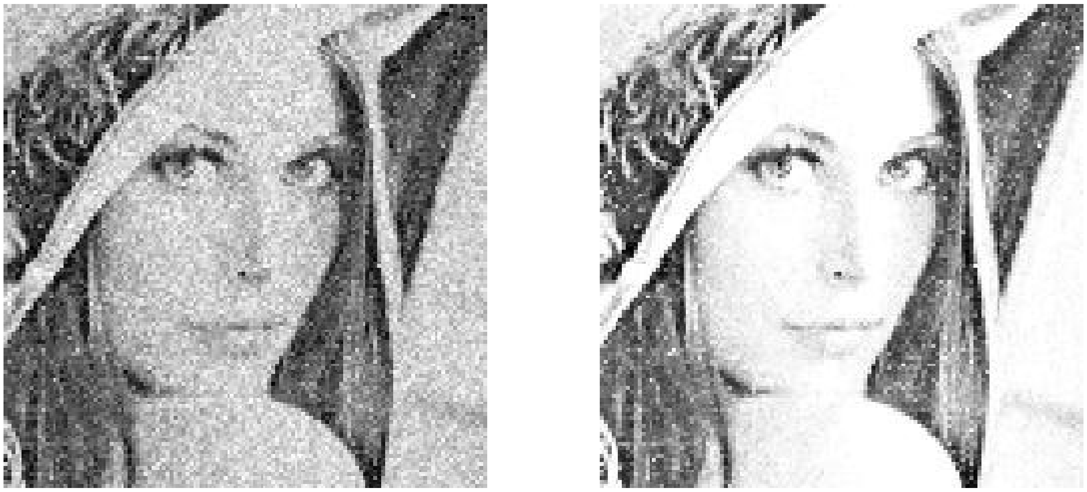
s is used as the input of the hybrid network in Fig. 2. In this experiment, the learning rate of the nonlinear module is $\mu(k) = 0.5 \times (0.9)^k$, λ in Eq. (14) is 0.5, and the number of hidden neurons in the nonlinear module, m, is 10. The fast ICA algorithm proposed by Hyvärinen and Oja[10] is used for the linear module. The ending condition is set as $e_2 < 0.1$. A typical nonlinearly-separated (restored) signal from the hybrid model, x, is shown in Fig. 7. For a comparison, the restored signal merely using the first sub-network, the restored signal merely using the second sub-network, and the restored signal using a simple linear ICA model (the fast ICA algorithm is used) are given in Figs. 8–10, separately. Also for a comparison, the linear ICA results (using the fast ICA algorithm) on the linear mixture in Fig. 5 are shown in Fig. 11. We have the following six observations:



(a) Channel 1

(b) Channel 2

Fig. 8. The restored signal x from the hybrid model using the first sub-network only.



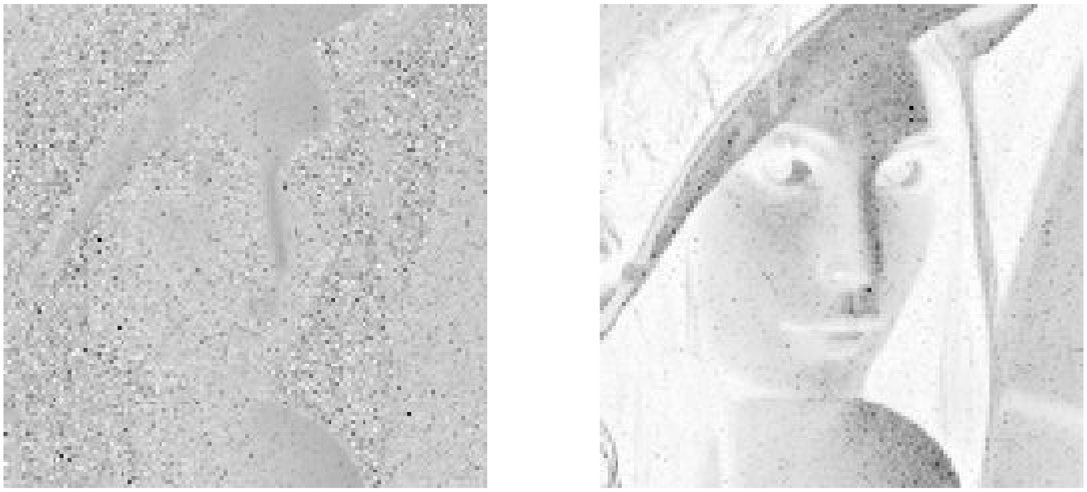
(a) Channel 1

(b) Channel 2

Fig. 9. The restored signal x from the hybrid model (using the second sub-network only).

(1) Nonlinear BSS problems exist widely. The images in Fig. 5 do not have visually significant difference from the images in Fig. 6, however, the former can be successfully (and easily) separated by a linear ICA algorithm (Fig. 11), but the latter can not be separated (just see the poor results in Fig. 10). Because of the many different possibilities for signal degrading in signal transmission, it is necessary to consider BSS as a nonlinear problem, but not a linear problem. This fact serves as the necessity of our nonlinear ICA model.

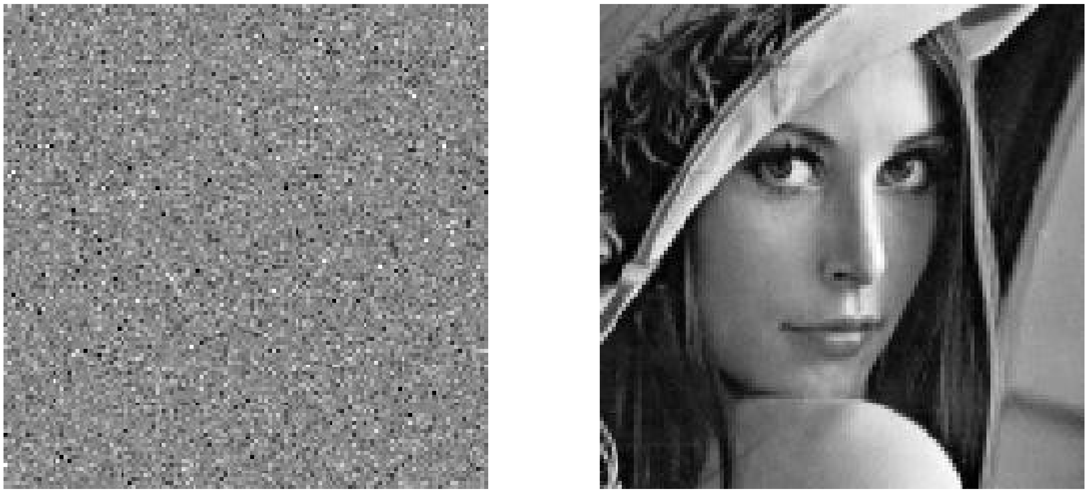
(2) The hybrid ICA model achieves better performance than the linear ICA model if observed signals are nonlinearly mixed. In Fig. 7, one channel of signal from the hybrid model is relatively clear image of Lenna while the other channel signal is mainly strong noise with the light shadow of Lenna. However, in Fig. 10, we cannot find any acceptable signals directly from a linear ICA model. Because the restored signals in Figs. 7 and 10 are not scaled properly, the signal-to-noise ratio (SNR) cannot be calculated immediately. However, the visual performance of the



(a) Channel 1

(b) Channel 2

Fig. 10. The restored signal x from a linear ICA model (using the fast ICA algorithm).



(a) Channel 1

(b) Channel 2

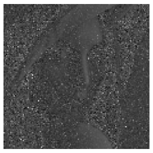
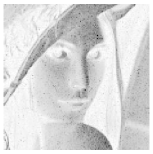
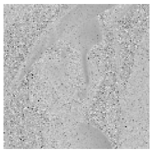
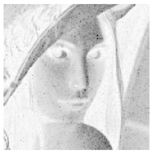
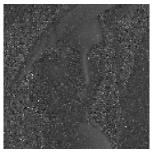
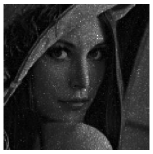
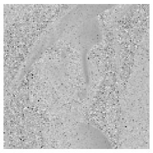
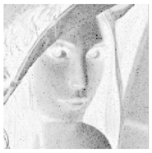
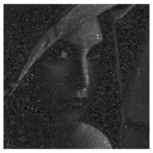
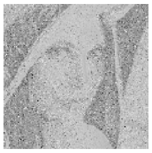
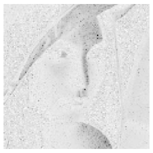
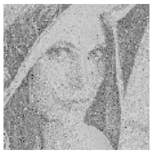
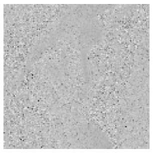
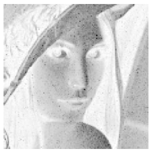
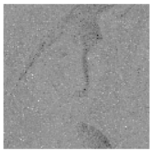
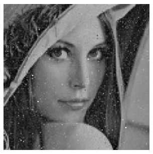
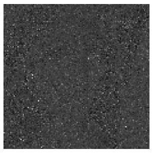
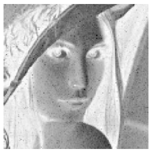
Fig. 11. The linearly separated signal from the linearly mixed signal h (by fast ICA algorithm).

hybrid model is undoubtedly better than that of the linear ICA model.

(3) The semi-parametric model achieves better performance than the parametric model and the non-parametric model. In Fig. 8, the linear parametric model with only the first sub-network fails to produce any visually acceptable signals. In Fig. 9, although both channels of output signal are visually fine, they are unacceptable logically because if the noisy channel of signal is not separated, the output signal is not useful. Therefore, the semi-parametric model is a necessary combination of the parametric model and the non-parametric model.

(4) The linear parametric model equals the linear ICA model. Apparently, Figs. 8 and 10 are identical. This situation is in accordance with our analysis in Sec. 2. Notice that when some forms of the post nonlinearity are known, parametric models other than

Table 2. An example of the batch learning.

Loop	e_2	$\|z\|$	x_1	x_2
1	1.000000	77.0376		
2	1.045722	544.8335		
3	0.393765	888.1439		
4	0.199703	740.3602		
5	0.436505	516.2926		
6	0.511347	1040.8		
7	0.158925	1234.9		
8	0.169648	1480.2		
9	0.055046	1408.2		

the linear channel model can be designed, in which cases the parametric model will not be identical to the linear ICA model.

(5) The original signal cannot be restored in this example. This point has been ignored in our previous work,[18] however, because the functional relationship between s_1 and h_1 is not reversible [Fig. 3 and Eq. (23)], there is actually no possibility to restore the original source signal in this example. When F is reversible, although there is no means to tell which solution is the original signal, the solution set does contain the original source signal. Thus an important aspect of our nonlinear ICA algorithm is that it can offer an acceptable solution even when the true solution does not exist.

(6) The converging speed of the hybrid model is quite fast. Typically, the batch algorithm in Table 1 will run about ten loops to meet the preset ending condition. That is, the nonlinear semi-parametric module is usually updated about ten times and converges. Because the fast ICA algorithm used in the linear ICA module is far (10–100 times) faster than the common gradient learning algorithms,[10] the whole batch learning procedure is faster than existing information back-propagation models.[8,9]

In addition to the above six observations, we have also examined the details of the batch learning. With a different mixing weight and different initial weights of the nonlinear module, the batch learning example in Table 2 finishes in nine steps. The index e_2 is shown in the second column of Table 2, where we see the convergence is fast, although not very stable. We also find the estimated signal, z, turns larger in the learning because its norm (shown in the third column in Table 2) turns larger. This indicates the effect of the second nonlinear sub-network is enlarged when the learning proceeds. The enlarged effect is mainly due to the enlarged weights of the nonlinear sub-network in the semi-parametric module. The fourth and fifth columns are the two channels restored signals. Notice that at loop 8 and loop 9, the restored signal is quite satisfying. (In fact such results are even visually better than that shown in Fig. 7. Note that the result of loop 9 has good visual quality when it is inverted. This type of intensity inversion is permitted in ICA.)

Besides the visual quality of the restored signal, is it possible to compute a quantitative index, say, the SNR, to give a convincing comparison on the image quality? The difficulty is due to that the absolute intensity difference between the ground truth signal and the restored signal is often large, even they have similar visual (subjective) quality. In our simulation, we use a binary image as one ground truth source and the impulsive noise as another ground truth source. Because the binary signal has simple intensity distribution, i.e., 1 for the foreground and -1 for the background, the subjective quality will agree with the measurable SNR in this case. The SNR is defined as:

$$SNR = \frac{\|g\|}{\|g - \hat{g}\|} \tag{24}$$

where g is the original signal and $\hat{g}$ is the estimation (or reconstruction) to g. The experimental results indicate an apparent improvement of SNR. For example, for the binary examplary image used in Peng and Chi,[18] we find the SNR of the hybrid model is 3.7941, which is much higher than the SNR of the linear model, which is only 0.0715. The result is consistent with the subjective quality reported by Peng and Chi.[18]

It should be pointed out again that generally the SNR is not a credible index for the nonlinear BSS problem, except in some special cases like binary sources. For this reason we consider following experiment 2 to investigate the ability of the hybrid model in nonlinear function approximation.

4.2. *Experiment 2: Nonlinear function approximation*

In experiment 1, because the functional relationship between h and s (denoted as G) is not reversible, the mapping from s to h is always different from the mapping from s to z. Actually it is impossible to find a one-to-one mapping from s to h. We therefore cannot examine whether the semi-parametric model can find the underlying nonlinear relationship between the observed signal and the independent ground truth sources (in the case they do not exist) or not, except the visual judgement on the restored signal. However, when the function G is reversible, the hybrid model should have the ability to find another nonlinear function F to approximate G. Of course the approximation is subject to some simple indeterminacies, such as the permutation,

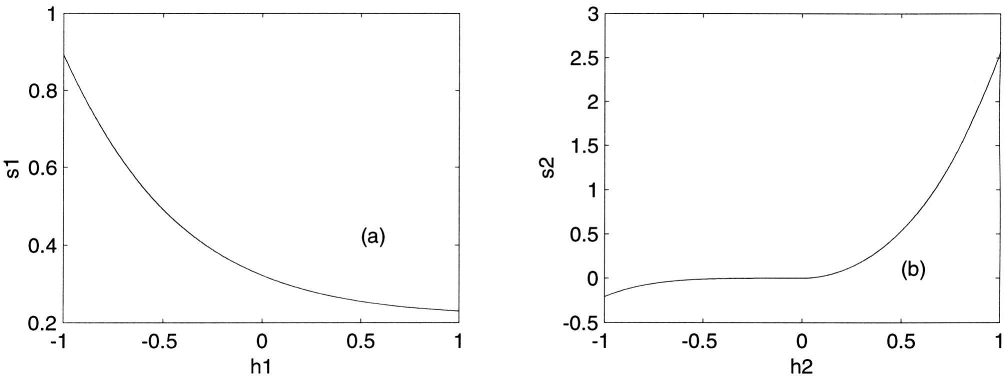
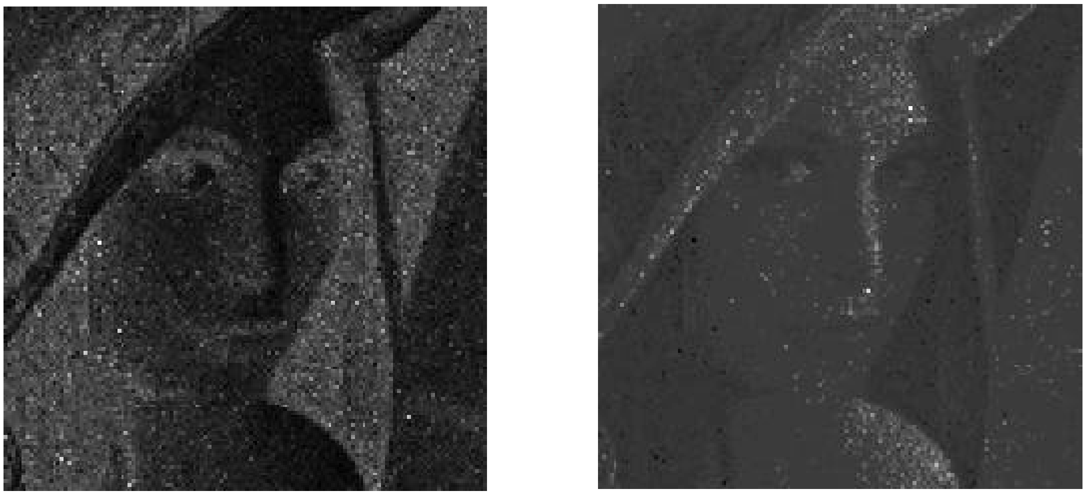


Fig. 12. Cross-channel nonlinear relationships of Eq. (25).



(a) Channel 1

(b) Channel 2

Fig. 13. Nonlinearly mixed signal s with Eq. (25).

amplitude, etc. But the basic shapes of F and G should be similar with each other.

Consider the cross-channel nonlinearity function G in Eq. (25). When h_1 and h_2 take the point series in Figs. 3(a) and (b), the nonlinearly transformed signals s_1 and s_2 have the shapes in Figs. 12(a) and (b), respectively. The first channel has a quick degrading. The second channel has a slow degrading when h is negative and a fast magnifying when h is positive.

$$\begin{cases} s_1 = 2\{\ln[\exp(4 - h_1 - h_2) + 1000] - 6.8\} \\ s_2 = \text{sgn}(h_1) \circ [\sqrt{|h_1 \circ h_2|} - \sinh(-h_2)] \circ \sinh(h_2) \end{cases} \tag{25}$$

With the same ground truth images "Lenna" and implusive noise and the same parameters in experiment 1, the nonlinear mixed signal s is produced and shown in Fig. 13. All the sample points are plotted in Fig. 14 to visualize the cross-channel nonlinearity. Figures 12 and 14 are similar, however, the hybrid model is fed with s and tries to learn the functional relationship shown in Fig. 14.

The results shown in Figs. 15 and 16 were obtained after the hybrid model converged after 7 loops. Figure 15 shows the results computed from the semi-parametric neural network. We see that the basic shape of G in Fig. 14 is well-reproduced in Figs. 15(a)–(d), where a possible sign change, shift

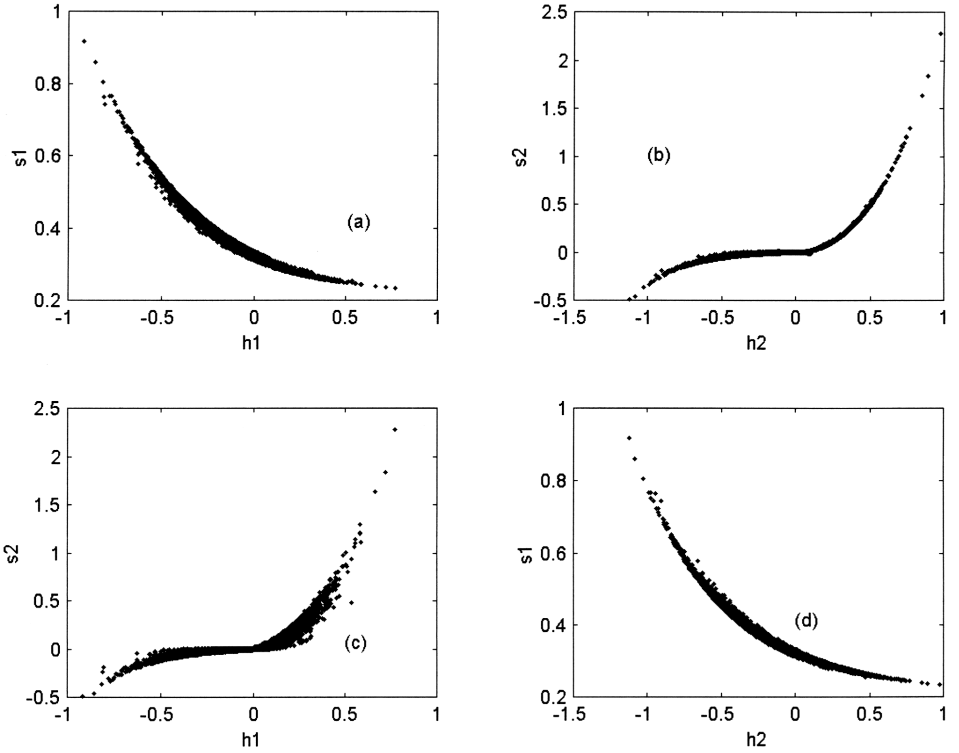


Fig. 14. Functional relationships between s and h.

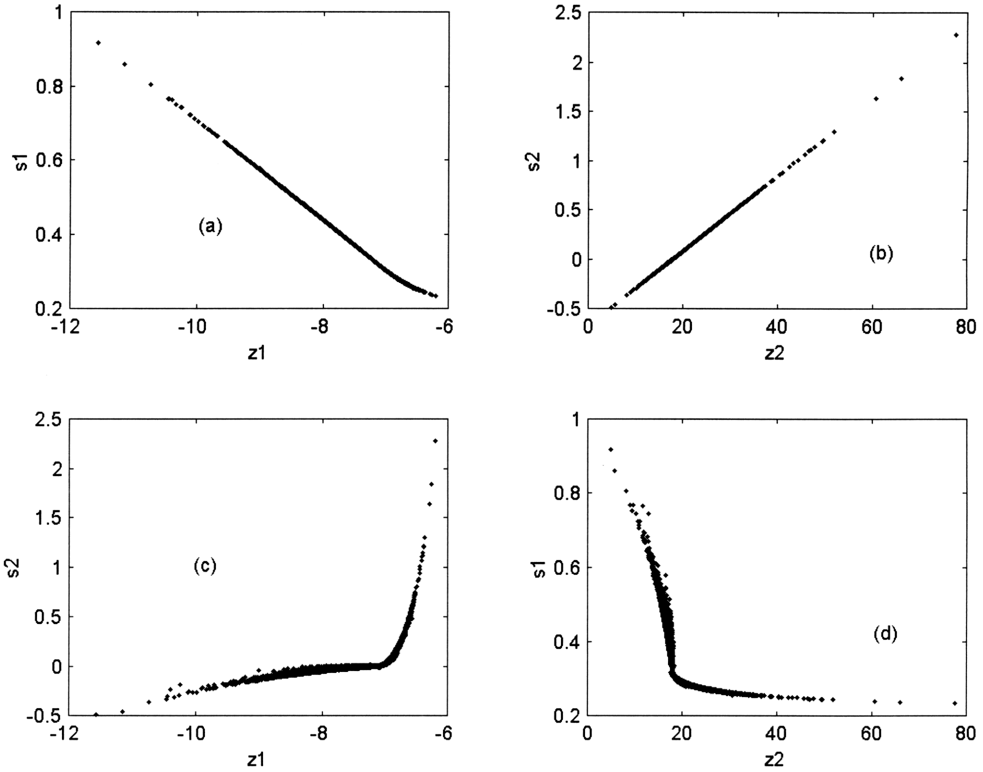
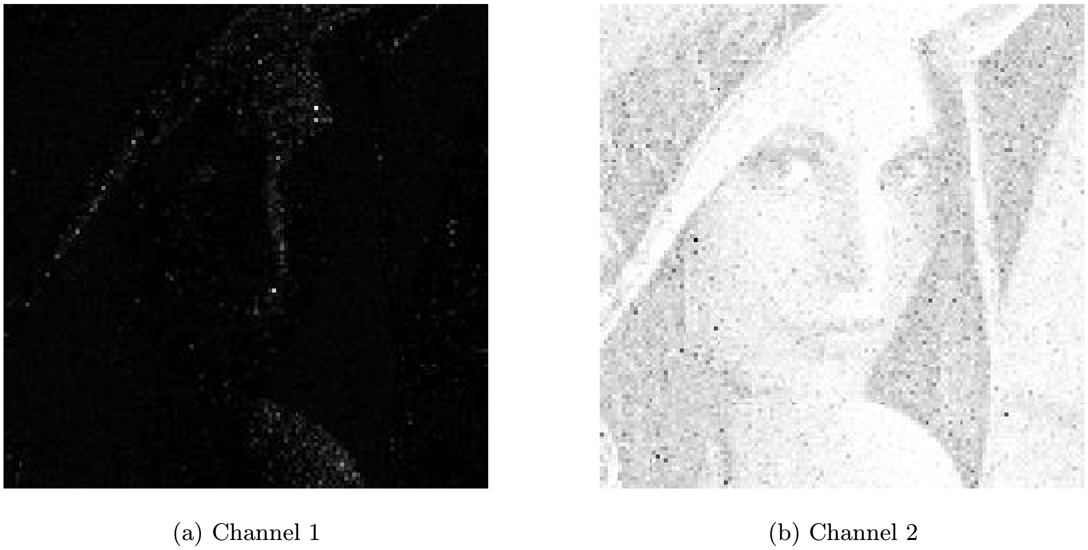


Fig. 15. Functional relationships between s and z.


(a) Channel 1

(b) Channel 2

Fig. 16. The restored signal x from the hybrid model.

and amplitude scaling are permitted. Particularly, the slow and fast varying areas in Figs. 14(a) and (c), the decreasing trend in Figs. 14(a) and (d), and the increasing trend in Figs. 14(b) and (c), are reproduced in Figs. 15(a) and (c), Figs. 15(a) and (d), and Figs. 15(b) and (c), separately.

Figure 16 shows the restored signal x. Similar to experiment 1, the hybrid model successfully construct an acceptable solution, where the main part of the first channel [Fig. 16(a)] is noise and the second channel [Fig. 16(b)] contains lightly degraded "Lenna" image. The restored signal is not as good as Fig. 7, perhaps because of the very poor observed signal in Fig. 13.

5. Conclusion

Semi-parametric models are arousing wider notice. In this paper, we propose a new semi- parametric model for nonlinear ICA. Because the cross-channel post nonlinear blind mixtures can lead to an infinite number of reasonable solutions, it is worth proposing the hybrid model for fast finding an acceptable compromise between the mutual independence of the output signals and the unknown post nonlinearity. It is also possible to present some assumptions on the post nonlinearity in order to achieve better results. In this paper, our semi-parametric hybrid model attempts to construct acceptable solutions of nonlinear BSS problems. This model can approach relatively strong nonlinearity from a linear model. We present the algorithm based on the entropy maximization and simplify it using a batch learning paradigm. Our simulation results confirm that this hybrid model can effectively produce a visual satisfying solution and a good approximation to the underlying nonlinearity. In binary cases, this model also produces the restored signal of a higher SNR than other models.

Acknowledgments

The work described in this paper was partially supported by a grant from the Hong Kong Polytechnic University (project no. A042, which is under the ASD of Multimedia Media Processing) and partially supported by a Chinese NSF Key Project (No. 6507032036) in Chien-Shiung Wu Laboratory, Department of Biomedical Engineering, Southeast University, Nanjing, China.

Appendix A. Derivation of Eq. (15)

Denote the gradient of C to z as ∇_z, from Eq. (14) we have,

$$\nabla_z = \frac{d\|z(k) - z(k-1)\|}{dz} - \lambda \frac{dH(y)}{dz} \tag{A-1}$$

Substitute Eq. (12) into Eq. (A-1), there is,

$$\begin{aligned}\nabla_z &= \frac{d\|z(k)-z(k-1)\|}{dz} - \lambda\frac{d}{dz}E\left[\ln\left|\prod_{i=1}^{n}(1-x_i)x_i\right|\right] \\ &= 2[z(k)-z(k-1)] - \lambda\frac{d}{dz}E\left[\ln\left|\prod_{i=1}^{n}(1-x_i)x_i\right|\right]\end{aligned} \tag{A-2}$$

We can impose the expectation operator on both terms in Eq. (A-2) and obtain the following approximation in Eq. (A-3), which is Eq. (15):

$$\begin{aligned}\nabla_z &\approx 2\{E[z(k)] - K[z(k-1)]\} \\ &\quad - \lambda\left\{W^T \times \frac{\Gamma_{n:1} - 2E[x]}{E[x]-E[x\circ x]}\right\}\end{aligned} \tag{A-3}$$

Appendix B. Derivation of Eqs. (16–20)

From Appendix A, the gradient of C to z has been obtained as ∇_z. The error back-propagation method is used to obtain the adjustment of nonlinear module parameters:

$$\Delta c = -\mu\nabla_z \circ s \tag{B-1}$$

$$\Delta v = -\mu\nabla_z \times o^T \tag{B-2}$$

$$\Delta v^{(b)} = \mu\nabla_z \tag{B-3}$$

$$\Delta u = -\frac{\mu}{2}\{v^T \times \nabla_z \circ (\Gamma_{m:1} - o\circ o)\} \times s^T \tag{B-4}$$

$$\Delta u^{(b)} = \frac{\mu}{2}\{v^T \times \nabla_z \circ (\Gamma_{m:1} - E[o\circ o])\} \tag{B-5}$$

In the batch algorithm, we impose the expectation operator on each of the above equations, and produce the approximation formulas in Eqs. (16–20) easily.

References

1. T. W. Lee 1998, *Independent Component Analysis: Theory and Applications* (Kluwer Academic Publishers).
2. A. Hyvärinen and P. Pajunen 1999, "Nonlinear independent component analysis: Existence and uniqueness results," *Neural Networks* **12**, 429–439.
3. G. Burel 1992, "Blind separation of sources: A nonlinear neural algorithm," *Neural Networks* **5**(6), 937–947.
4. G. Deco and W. Brauer 1995, "Nonlinear higher-order statistical decorrelation by volume-conserving neural networks," *Neural Networks* **8**, 525–535.
5. T. W. Lee, B. Koehler and R. Orglmeister 1997, "Blind source separation of nonlinear mixing models," *1997 IEEE Int. Workshop on Neural Networks for Signal Processing*, Florida, 406–415.
6. P. Pajunen, A. Hyvärinen and J. Karhunen 1996, "Nonlinear blind source separation by self-organizing maps," *Proc. of the 1996 Int. Conf. on Neural Information Processing*, Hong Kong, 1207–1210.
7. P. Pajunen and J. Karhunen 1997, "A maximum likelihood approach to nonlinear blind source separation," *Proc. of the 1997 Int. Conf. on Artificial Neural Networks*, Lausanne, Switzerland, 541–546.
8. A. Taleb and C. Jutten 1997, "Nonlinear source separation: The post-nonlinear mixtures," *Proc. of ESANN'97*, Bruges, Belgium, 279–284.
9. H. H. Yang, S. Amari and A. Cichocki 1998, "Information-theoretic approach to blind separation of sources in nonlinear mixture," *Signal Processing* **64**(3), 291–300.
10. A. Hyvärinen and E. Oja 1997, "A fast fixed-point algorithm for independent component analysis," *Neural Computation* **9**(7), 1483–1492.
11. A. J. Bell and T. J. Sejkowski 1995, "An information maximization approach to blind separation and blind deconvolution," *Neural Computation* **7**, 1129–1159.
12. J. N. Kapur and H. K. Kesavan 1992, *Entropy Optimization Principles with Applications* (Academic Press, Boston).
13. M. Girolami 1997, Self-organizing artificial neural networks for signal separation. Ph.D. Dissertation, Department of Computing and Information Systems, Paisley University, Scotland.
14. B. Pearlmutter and L. Parra 1996, "A context-sensitive generalization of ICA," *ICONIP'96*, 151–157.
15. J.-F. Cardoso 1997, "Infomax and maximum likelihood for blind source separation," *IEEE Signal Processing Letters* **4**, 109–111.
16. P. Comon, 1994, "Independent component analysis — a new concept?" *Signal Processing* **36**(3), 287–314.
17. M. Girolami and C. Fyfe 1997, "Stochastic ICA contrast maximization using Oja's nonlinear PCA algorithm," *International Journal of Neural Systems* **8**(5 & 6), 661–678.
18. H.-C. Peng and Z. Chi 1999, "Approaching the post non-linearity of blind mixtures by hybrid neural network," *Proc. of the 1999 Int. Joint Conf. of Neural Networks*, Washington, No. 2125.